IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

TITLE

DETERMINING POWER CONSUMPTION OF AN APPLICATION

INVENTOR:
NIGEL C. PAVER

PREPARED BY:

LIBBY H. HOPE
PATENT ATTORNEY
INTEL CORPORATION
2200 MISSION COLLEGE BOULEVARD
SANTA CLARA, CA 95052
(408) 765-8080

EXPRESS MAIL CERTIFICATE OF MAILING
"Express Mail" Mailing No.: EV374754029US

I hereby certify that this paper is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 C.F.R. §1.10 on the date indicated below and that this paper has been addressed to Mail Stop Patent Application, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

Date of Deposit: March 31, 2004

Name of Person Mailing Correspondence: Debbie Peloquin

Debbie Peloquin March 31, 2004
Signature Date

DETERMINING POWER CONSUMPTION OF AN APPLICATION

FIELD

[0001] Embodiments of this invention relate to determining power consumption of an application.

BACKGROUND

[0002] Under the current state of the art, power consumption properties of a processor may be measured to, for example, prevent the silicon device (hereinafter "chips") on which the processor resides from overheating. One example of a power consumption property is temperature. For example, chips may have on-chip temperature sensors that may each include a number of outputs that indicate a current temperature (or temperature range) of the chip. Since the efficiency at which an application executes may be related to how much power is consumed by the application code, it may be useful to know which parts of the application consume the most power so that the application code may be optimized.

BRIEF DESCRIPTION OF THE DRAWINGS

[0003] Embodiments of the present invention are illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

[0004] FIG. 1 illustrates a system according to first embodiment.

[0005] FIG. 2 illustrates a system according to a second embodiment.

[0006] FIG. 3 is a flowchart illustrating a method according to one or more embodiments.

[0007] FIG. 4 is a flowchart illustrating a method according to one or more embodiments.

DETAILED DESCRIPTION

[0008] Examples described below are for illustrative purposes only, and are in no way intended to limit embodiments of the invention. Thus, where examples may be described in detail, or where a list of examples may be provided, it should be understood that the examples are not to be construed as exhaustive, and do not limit embodiments of the invention to the examples described and/or illustrated. Furthermore, illustrated embodiments are intended to illustrate an example of an embodiment, and should not be construed as limiting embodiments of the invention.

[0009] FIG. 1 illustrates a system 100 that may be used in one embodiment of the invention, and FIG. 2 illustrates a system 200 that may be used in another embodiment of the invention. System 100 may comprise a first node 150A, and a second node 150B, where first node 150A and second node 150B may communicate via connections 106A, 106B to a network 106. System 200 may comprise a first node 250.

[0010] System 100, 200 may comprise circuitry 126 to perform one or more operations described herein. Circuitry 126 may be embodied in hardware and be hardwired to perform the one or more operations. Some or all of circuitry 126 may be comprised in, for example, processor 102, 202 in other structures, systems, and/or devices that may be, for example, comprised in a motherboard (not illustrated), and/or communicatively coupled to bus 106, and may exchange data and/or commands with one or more other components in system 100, 200.

Such other components may comprise, for example, memory, and/or processor 102, 202. Many possibilities exist; however, not all possibilities may be illustrated.

[0011] Additionally or alternatively, circuitry 126 may be embodied in software to execute machine-executable instructions to perform these operations. For example, circuitry 126 may comprise memory that may store machine-executable instructions that may be executed by circuitry 126 to perform these operations. Additionally or alternatively, circuitry 126 may access machine-executable instructions from one or more other memories, such as memory.

[0012] Circuitry 126 may comprise, for example, one or more digital circuits, one or more analog circuits, one or more state machines, programmable circuitry, and/or one or more ASIC's (Application-Specific Integrated Circuits). Additionally or alternatively, circuitry 126 carrying out some or all of the operations described herein may result from the execution of program instructions and/or the accessing, operation upon, and/or manipulation of data by circuitry 126.

[0013] FIG. 3 illustrates a method according to at least one embodiment of the invention. The method begins at block 300 and continues to block 302 where circuitry 126 may monitor one or more sensor outputs 108A, 108B, 108C, 108D of sensor 112. Sensor 112 may be on chip 110, 210. Alternatively and/or alternatively, sensor 112 may be part of a component external to chip 110, 210 such as, for example, a power management integrated circuit, or a power supply that may determine property being consumed by chip 110, 210.

[0014] Sensor 112 may measure a power consumption property of chip 110, 210 and each sensor output 108A, 108B, 108C, 108D may indicate the existence of the power consumption property. In one embodiment, the power consumption property may comprise temperature, and each sensor output 108A, 108B, 108C, 108D may indicate a temperature that corresponds to one or more temperatures of the chip 110, 210. A temperature may refer to a single temperature, or one or more ranges of temperatures. Therefore, each sensor output 108A, 108B, 108C, 108D may indicate that the power consumption property of the chip 110, 210 is within a range of measurements of the power consumption property corresponding to the given sensor output 108A, 108B, 108C, 108D. Other power consumption properties may be measured. For example, the measured power consumption property may be voltage drop (IR).

[0015] In one embodiment, chip 110, 210 may comprise processor 102, 202. Processor 102, 202 may comprise, for example, an Intel® XScale® microprocessor (hereinafter "XScale®") that is commercially available from the Assignee of the subject application. Of course, alternatively, processor 102, 202 may comprise another type of microprocessor, such as, for example, a microprocessor that is manufactured and/or commercially available from a source other than the Assignee of the subject application, without departing from this embodiment.

[0016] At block 304, circuitry 126 may record a time that each of the one or more sensor outputs 108A, 108B, 108C, 108D indicates the existence of the power consumption property at a corresponding measurement. For example, if sensor indicates a certain power consumption property (e.g., temperature) at a certain measurement (e.g., 75°C), such indication may be communicated to a corresponding one of the sensor outputs 108A, 108B, 108C, 108D. The corresponding sensor output 108A, 108B, 108C, 108D, therefore, may indicate the presence of the power consumption property by indicating the measurement or a range of the measurement for the power consumption property of chip 110, 210. The recorded time may be cumulated.

[0017] Circuitry 126 may monitor one or more sensor outputs 108A, 108B, 108C, 108D for such indication, and record the time that the one or more monitored sensor outputs 108A, 108B, 108C, 108D indicates the presence of the power consumption property at corresponding measurements. The time may be stored in a database, for example. The time may additionally be outputted, and/or accessed. Additionally, time may be provided in various metrics, including milliseconds, seconds, percentages, number of times the sensor outputs indicates the power consumption property, or any combination of these or other metrics not mentioned, for example.

[0018] The method ends at block 306.

[0019] In one embodiment, circuitry 126 incorporating the method described above may be embodied in an event counter 114. For example, event counter 114 may be provided by the Intel® XScale® microarchitecture (hereinafter "XScale®") that is commercially available from the Assignee of the subject application. In this embodiment event counter 114 may monitor sensor outputs 108A, 108B, 108C, 108D of sensor 112. Event counter 114 may additionally monitor and count other events, such as pipeline stalls and cache misses.

[0020] FIG. 4 illustrates a method that may employ system 100, 200 in at least one embodiment. The method begins at block 400 and continues to block 402 where circuitry 126 may obtain event data, where the event data includes a time that each of one or more sensor outputs indicates an existence of a power consumption property at a corresponding measurement. As used herein, "event data" refers to data that is measured by a counter. In one embodiment, counter may comprise event counter 114, and event data may comprise the amount of time that each of sensors 108A, 108B, 108C, 108D indicates the presence of a power consumption property at a corresponding measurement. Other examples of event data may include number of cache misses, number of pipeline stalls, and that sensor outputs indicate the existence of a corresponding temperature range. Where the event data is the existence of a corresponding temperature range, for example, the "corresponding measurement" refers to the existence of a temperature range for a given sensor output.

[0021] At block 404, circuitry 126 may monitor one or more parts of one or more applications 120. As used herein, an "application" refers to a computer-accessible program that may comprise a plurality of instructions, such as machine-executable instructions. The plurality of instructions may be stored in a memory. Application 120 may comprise an entire set of instructions that make up a program, or it may comprise a subset of the instructions that make up a program. For example, application 120 may refer to one or more modules of the program. Furthermore, a part of an application 120 may refer to one or more lines of code in the application 120, or a specific module of the application, for example.

[0022] At block 406, circuitry 126 may correlate the one or more parts of application 120 with the power consumption property read from sensor outputs 108A, 108B, 108C, 108D. The correlation of information may be outputted, and/or accessed.

[0023] The method ends at block 408.

[0024] In one embodiment, circuitry 126 incorporating the method described above may be embodied in performance analyzer 116, 216. As used herein, a "performance analyzer" refers to an application that may use tools, such as a performance collector, to collect application information and correlate the information with event data. The correlated information may be used to analyze the performance of an application, such as application120, where the application may comprise the performance analyzer 116, 216 itself, or one or more other applications 120. The performance analyzer 116, 216 may use the correlated information to provide recommendations for optimizing performance of the application 120. For example, the Intel® VTune™ Performance Analyzer (hereinafter "VTune™") is an example of a performance analyzer 116, 216 that may analyze an application 120 that may run on an XScale®-based processor 102, 202 such as an Intel® PXA25x or Intel® PXA26X processor. VTune™ may include one or more performance collectors 118, 218 to identify potential performance issues, and to provide recommendations for improving the application's 120 performance.

[0025] In VTune™, for example, active instruction addresses may be collected for analysis, where the collection of such data may be referred to as sampling. In VTune™, sampling may be accomplished by occasionally interrupting normal system execution to sample address instructions. An interrupt may be triggered after a preprogrammed number of events has occurred. Instruction addresses may be sampled, and then correlated with event data. For example, each time sensor output 108A may indicate a corresponding temperature range (or voltage drop) during execution of application 120, event counter 114 may be incremented (or may start a timer that tracks the amount of time at the temperature range).

[0026] When the event counter 114 reaches a specified value (e.g., time indicated by a sensor output exceeds 100 milliseconds), performance collector 118, 218 may sample an instruction address at the part of the application 120 currently being executed. The instruction address may correspond to one or more lines of code, and/or a specific module of the application. Performance collector 118, 218 may correlate the sampled instructions to a corresponding power consumption property. The executed instruction and power consumption property may then be recorded and/or outputted by performance analyzer 116, 216, for example, and be used to optimize performance and power consumption of the application 120.

[0027] In one embodiment, as illustrated in FIG. 1, performance analyzer 116 may be executed by processor 104 on a node different from the node on which application 120 being analyzed is executed. In this embodiment, processor 104 may comprise, for example, an Intel® Pentium® microprocessor that is commercially available from the Assignee of the subject application. In another embodiment, as illustrated in FIG. 2, performance analyzer 216 may be executed by processor 202 on the same node on which application 120 being analyzed may be executed. In this embodiment, processor 202 may comprise an Intel® XScale® microprocessor, for example.

[0028] Event counter 114, performance collector 118, 218, and performance analyzer 116, 216 may be comprised in software, such as machine-executable instructions that may be executed by circuitry, such as circuitry 126, of processor 102, 202. Of course, these programs may alternatively be comprised in firmware or in hardware. Also, machine-executable instructions may be stored in a memory (not shown) and executed by circuitry 126, such as in processor 102, 202.

[0029] Embodiments of the present invention may be provided, for example, as a computer program product which may include one or more machine-accessible media having machine-executable instructions that, when executed by one or more machines such as a computer, network of computers, or other electronic devices, may result in the one or more machines carrying out operations in accordance with embodiments of the present invention. A machine-accessible medium may include, but is not limited to, floppy diskettes, optical disks, CD-ROMs (Compact Disc-Read Only Memories), magneto-optical disks, ROMs (Read Only Memories), RAMs (Random Access Memories), EPROMs (Erasable Programmable Read Only Memories), EEPROMs (Electrically Erasable Programmable Read Only Memories), magnetic or optical cards, flash memory, or other type of media / machine-readable media suitable for storing machine-executable instructions.

[0030] Moreover, embodiments of the present invention may also be downloaded as a computer program product, wherein the program may be transferred from a remote computer (e.g., a server) to a requesting computer (e.g., a client) by way of one or more data signals embodied in and/or modulated by a carrier wave or other propagation medium via a communication link (e.g., a modem and/or network connection). Accordingly, as used herein, a machine-readable medium may, but is not required to, comprise such a carrier wave.

Conclusion

[0031] Therefore, in one embodiment, a method may comprise monitoring one or more sensor outputs of a sensor, the sensor to measure a power consumption property of the chip, and each sensor output to indicate a measurement of the power consumption property; and recording a time that each of the one or more sensor outputs indicates an existence of the power consumption property at the measurement corresponding to each of the one or more sensor outputs.

[0032] Embodiments of the invention may enable a system to determine how much time is spent at certain a certain power consumption property, such as temperature. In one embodiment, this information may be correlated with sampled instructions addresses in an application to determine how much power is consumed by various parts of the application. The correlated information may be used to optimize the application code.

[0033] In the foregoing specification, embodiments of the invention have been described. It will, however, be evident that various modifications and changes may be made to these embodiments without departing therefrom. The specifications and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense.